**VERBALE RIUNIONE 7: 16/11**

**Presenti:** Facci Matteo, Guidobene Davide, Pittini Enrico, Rosin Giacomo

Il documento di Progettazione

1. Introduzione. Scopo del documento. Descrizione sintetica delle funzionalità più importanti.

2. Architettura del Sistema. Descrizione sottosistemi e moduli. Metodo di progettazione (oggetti/funzionale). Presentare possibili soluzioni diverse.

3. Modello dei Dati e del Controllo. Come sono organizzati i dati e il controllo all’interno della nostra architettura.

Modello dei dati → Client-Server

Modello del controllo → Basata su eventi-interrupt driver

4. Modelli UML: diagrammi delle classi e di sequenza.

* Diagrammi classi del sistema (architettura del sistema)
* Diagrammi sequenze per ogni singolo caso d’uso.

5. Progettazione dell’interfaccia Utente

6. Glossario

2. Architettura del Sistema.

Possibili sottosistemi

* Inizializzazione comunicazione (creazione/accesso stanza virtuale)
* Gestione comunicazione
* Gestione tavoli (lista tavoli occupati/liberi)
* Gestione sensori (rilevazione shake, gps)
* Gestione informazioni locale
* Gestione identificazione/autenticazione e gestione modalità applicazione

Modello ad oggetti. Sistema decentralizzato con oggetti che si passano il controllo.

(Cenni su considerazioni relative a qualità ~ accoppiamento, coesione, …)

Mostrare varie alternative :

* unico sottosistema comunicazione/diviso in 2 → abbiamo scelto il secondo per maggiore coesione e accoppiamento

3. Modello dei Dati e del Controllo.

* Modello dei dati → Client-Server

Non si sono dati centralizzati ma sono tutti interni ai vari sottosistemi. Non c’è necessità o utilità nel tenere dati centralizzati.

* Modello del controllo → Basata su eventi-interrupt driver.

Programma organizzato in modo massiccio ad eventi. Quando accade un evento sono chiamati solo i sottosistemi associati.

4. Modelli UML: diagrammi delle classi e di sequenza.

A Diagrammi delle classi.

Possibili classi:

* Locale. Classe che rappresenta l’intero locale ed è in pratica il programma. Al suo interno contiene un oggetto (singleton) di Stanza virtuale, di Tavoli, di Notifiche Shake.
* Stanza virtuale (inizializzazione comunicazione)

- Costruttore (chiamato dal gestore)

- Avviamento (gestore)

- Distruttore (gestore)

- Partecipa (cliente)

- Togli(cliente)

Possibile idea. Oggetto “singleton” della classe stanza virtuale creato dal gestore. Gestore applica metodo per avviare stanza virtuale(bluetooth,wifi,..). Cliente ,tramite classe che si occupa della comunicazione, notifica il gestore della sua volontà di unirsi alla stanza virtuale: il gestore applica metodo partecipa all’oggetto.

* Gestione comunicazione (Nearby)

Possibili sottoclassi cliente/gestore

Metodi sottoclasse cliente:

* connetti()